

**Advanced Go Programming**

**Course Number:** GO-116
**Duration:** 3 days

**Overview**

This Advanced Go Programming training course teaches attendees the patterns, tools, and strategies needed to solve complex problems in Golang. This Go course also teaches how to evaluate design and implementation decisions by systematically profiling applications.

**Prerequisites**

All attendees should have taken Accelebrate's [Building Applications in Go](file:////training/go-programming-applications) or have the equivalent knowledge.

**Materials**

All Golang training students receive comprehensive courseware.

**Software Needed on Each Student PC**

* The latest distribution of Go for your operating system (Windows, Mac, or Linux)
* A Go-compatible IDE such as JetBrains GoLand
* Related lab files that Accelebrate provides

**Objectives**

* Learn and apply some common concurrency patterns
* Build RESTful services using libraries and frameworks
* Apply effective testing strategies
* Profile go applications to fix performance bottlenecks
* Evaluate and implement Microservices communication using gRPC
* Create resource-efficient Docker images for Go applications

**Outline**

* Golang Overview
	+ Language constructs
	+ Structs, methods, and interfaces
	+ Error handling
	+ Concurrency basics
* Restful Services
	+ Building HTTP services
	+ Framework choices
	+ Gin framework
	+ Handling JSON
	+ Versioning APIs
	+ Middleware
	+ Authentication and authorization
* Advanced Concurrency
	+ Advanced concurrency patterns
	+ Runner
	+ Worker
	+ Pools
	+ Signals
	+ Migrations
* Testing
	+ Testing
	+ Micro benchmark
* Profiling
	+ Scheduler
	+ GC
	+ Escape analysis
	+ Profiling
	+ Tracing
* Code Generation
	+ Templating
	+ Generators
* gRPC Service
	+ Protocol buffers
	+ Communication patterns in gRPC
	+ Request and response
	+ Client streaming
	+ Server streaming
	+ Bidirectional streaming
	+ Interceptors
* Web Sockets
	+ Web socket packages in Go
	+ Creating real-time services
* Dockerizing Go Applications
	+ Provisioning images for Go applications
	+ Environment variables
	+ Installing dependencies
	+ Creating a build
* Conclusion